

## DEGREE CORRELATION OF BIPARTITE NETWORK ON PERSONALIZED RECOMMENDATION

JIAN-GUO LIU\*, TAO ZHOU<sup>†</sup>, BING-HONG WANG and YI-CHENG ZHANG

*Research Center of Complex Systems Science  
University of Shanghai for Science and Technology  
Shanghai 200093, P. R. China*

*Department of Modern Physics  
University of Science and Technology of China  
Hefei 230026, P. R. China*

*Department of Physics, University of Fribourg  
Fribourg CH-1700, Switzerland*

*\*liujg004@ustc.edu.cn*

*<sup>†</sup>zhutou@ustc.edu*

QIANG GUO

*Business School, University of Shanghai for Science and Technology  
Shanghai 200093, P. R. China*

In this paper, the statistical property, namely degree correlation between users and objects, is taken into account and be embedded into the similarity index of collaborative filtering (CF) algorithm to improve the algorithmic performance. The numerical simulation on a benchmark data set shows that the algorithmic accuracy of the presented algorithm, measured by the average ranking score, is improved by 18.19% in the optimal case. The statistical analysis on the product distribution of the user and object degrees indicate that, in the optimal case, the distribution obeys the power-law and the exponential is equal to  $-2.33$ . Numerical results show that the presented algorithm can provide more diverse and less popular recommendations, for example, when the recommendation list contains 10 objects, the diversity, measured by the hamming distance, is improved by 21.90%. Since all of the real recommendation data evolving with time, this work may shed some light on the adaptive recommendation algorithm which could change its parameter automatically according to the statistical properties of the user-object bipartite network.

*Keywords:* Recommendation systems; bipartite network; collaborative filtering.

## 1. Introduction

With the expansion of Internet<sup>1</sup> and widely applications of *Web 2.0*, how to efficiently help people obtain information that they truly need is a challenging task nowadays.<sup>2</sup> Recommender systems have become an effective tool to address the information overload problem by predicting the user's interests and habits based on their historical selections or collections, which have been used to recommend books and CDs at Amazon.com, movies at Netflix.com, and news at Versifi Technologies (formerly AdaptiveInfo.com).<sup>3</sup> Motivated by the practical significance to the e-commerce and society, study of recommender systems has caught increasing attentions and become an essential issue in Internet applications such as e-commerce systems and digital library systems.<sup>4</sup> A personalized recommender system includes three parts: data collection, model analysis and the recommendation algorithm, among which the algorithm is the core part. Various kinds of algorithms have been proposed thus far, including collaborative filtering approaches,<sup>5–10</sup> content-based analyses,<sup>11–14</sup> hybrid algorithms,<sup>15–17</sup> and so on. Although the personalized recommender algorithms have been extensively studied, most of them are proposed from the viewpoints of computer science and e-commerce, in which only focus on the accuracy on a static data set. In fact, all of the real recommendation data, constructed by the user-object bipartite network, are evolving simultaneously, therefore, the performance of the existed algorithms could not be guaranteed. In order to present an adaptive algorithm, the correlation between the statistical properties of system data and the algorithm performance must be studied from the viewpoint of statistical physics.

One of the most successful recommendation algorithms, called *collaborative filtering* (CF), has been developed and extensively investigated over the past decade.<sup>5,6,18</sup> When predicting the potential interests of a given user, CF algorithm firstly identifies a set of similar users from the past records and then makes predictions based on the weighted combination of those similar users' opinions. Recently, some physical dynamics, including mass diffusion<sup>19,20</sup> and heat conduction,<sup>21</sup> have found their applications in CF algorithm. Liu *et al.*<sup>7</sup> introduced the mass diffusion process to compute the user similarity of CF, and found that the modified algorithm has remarkably higher accuracy than the standard CF. By considering the high-order correlation of the users and object, Liu *et al.*<sup>8</sup> and Zhou *et al.*<sup>22</sup> proposed the ultra-accuracy algorithms, in which the accuracy, measured by the average ranking score, could be improved to around 0.08. It should be noticed that these two algorithms are implemented based on the second-order user or object correlations, which hinder their application because of the limited computation resource. Moreover, the performances of these algorithms could be guaranteed to other testing data sets because the statistical properties of the testing data have not been taken into account. In the above methods, all of the objects and users with far different degrees have been treated equally, in other words, the degree correlations between objects and users are neglected. For example, suppose a user with small degree has

collected a small-degree object, the edge connecting them represents a very special taste of the user, while the information contained in the edges connecting an active user and a popular object is less meaningful. Therefore, we argue that the user similarity index could be improved by considering the degree correlation of the user-object bipartite network. The numerical results show that the improved index that depresses the influence of mainstream preferences can provide more accurate and more diverse recommendations.

## 2. Construction of the Collaborative Filtering Algorithm

Suppose there are  $m$  objects and  $n$  users in a recommender system. Denote the object set as  $O = \{o_1, o_2, \dots, o_m\}$  and the user set as  $U = \{u_1, u_2, \dots, u_n\}$ , a recommender system can be fully described by an adjacent matrix  $A = \{a_{ij}\} \in R^{m,n}$ , where  $a_{ij} = 1$  if  $o_i$  is collected by  $u_j$ , and  $a_{ij} = 0$  otherwise. In the standard CF, the user or object similarities are calculated first, then the predictions are computed accordingly. If  $u_i$  has not yet collected  $o_j$  (i.e.  $a_{ji} = 0$ ), the predicted score,  $v_{ij}$ , is given as

$$v_{ij} = \frac{\sum_{l=1}^n s_{li} a_{jl}}{\sum_{l=1}^n s_{li}}, \quad (1)$$

where  $s_{li}$  is the similarity between user  $u_l$  and  $u_i$ . There are at least two ways previously proposed to measure similarity, as:

$$s_{ij} = \frac{2 \sum_{l=1}^m a_{il} a_{jl}}{k_{u_i} + k_{u_j}}, \quad (2)$$

$$s_{ij} = \frac{\sum_{l=1}^m a_{il} a_{jl}}{\sqrt{k_{u_i} k_{u_j}}}. \quad (3)$$

The Eq. (2) is called Sorensens index of similarity,<sup>23</sup> which was proposed by Sorensen in 1948; the Eq. (3), called the cosine similarity, was proposed by Salton in 1983 and has a long history of the study on citation networks<sup>24</sup>; to a target user  $u_i$ , the algorithm is given as following:

- (i) Calculating the user similarity matrix  $\{s_{ij}\} \in R^{n,n}$ ;
- (ii) For each user  $u_i$ , according to Eq. (1), calculating the predicted scores for his/her uncollected objects;
- (iii) Sorting the uncollected objects in descending order of the predicted scores, and those objects in the top will be recommended.

## 3. Algorithmic Performance Metrics

To test a recommendation method on a dataset we randomly remove 10% of the links as the probe set and apply the algorithm to the remainder (training set) to produce a recommendation list for each user. We then employ three different metrics, one to measure accuracy in recovery of deleted links and two to measure recommendation popularity and diversity.

### 3.1. Average ranking score

An accurate method will put preferable objects in higher places. The average ranking score is adopted to measure the accuracy, which is defined as follows. For an arbitrary user  $u_i$ , if the entry  $u_i - o_j$  is in the probe set (according to the training set,  $o_j$  is an uncollected object for  $u_i$ ), we measure the position of  $o_j$  in the ordered list. For example, if there are  $L_i = 10$  uncollected objects for  $u_i$ , and  $o_j$  is the 3rd from the top, we say the position of  $o_j$  is 3/10, denoted by  $r_{ij} = 0.3$ . Since the probe entries are actually collected by users, a good algorithm is expected to give high recommendations to them, leading to small  $r_{ij}$ . Therefore, the mean value of the position,  $\langle r \rangle$ , averaged over all the entries in the probe, can be used to evaluate the algorithmic accuracy: the smaller the average ranking score, the higher the algorithmic accuracy, and vice versa.

### 3.2. Popularity and diversity

Besides accuracy, the average degree of all recommended objects,  $\langle k \rangle$ , and the mean value of Hamming distance,  $S$ , are taken into account to measure the algorithmic popularity and diversity.<sup>25</sup> The smaller average degree, corresponding to the less popular objects, are preferred since those lower-degree objects are hard to be found by users themselves. For example, suppose there are 10 perfect movies not yet known for user  $u_i$ , seven of which are widely popular, while the other three fit a certain specific taste of  $u_i$ . An algorithm recommending the seven popular movies is very nice for  $u_i$ , but he may feel even better about a recommendation list containing those two unpopular movies. In addition, the personalized recommendation algorithm should present different recommendations to different users according to their tastes and habits. The diversity can be quantified by the average Hamming distance,  $S = \langle H_{ij} \rangle$ , where  $H_{ij} = 1 - Q_{ij}/L$ ,  $L$  is the length of recommendation list and  $Q_{ij}$  is the overlapped number of objects in  $u_i$  and  $u_j$ 's recommendation lists. The largest  $S = 1$  indicates the recommendations to all of the users are totally different, in other words, the system has highest diversity. While the smallest  $S = 0$  means that the recommendations for different users are exactly the same.

## 4. Effect of Degree Correlation Between Users and Objects to CF

In the proposed CF algorithms, they only rely on the users' degrees and the number of common collected objects, without consideration of the influence of degree correlation between users and objects. Inspired by the mass diffusion process proposed by Zhou *et al.*,<sup>20</sup> Liu *et al.*<sup>7</sup> proposed a modified CF to improve the algorithmic accuracy by using the mass diffusion process to compute the user similarities, and they found that the diversity of recommendations is also enhanced. Although this algorithm has improved the standard CF, however, the degree correlation between users and objects has not been considered, thus every edge has the same contribution to the diffusion process. If both of  $u_i$  and  $u_j$  have selected an object  $o_l$ , they

probably have similar tastes or interests. Provided the degree of  $o_l$  is very large (object  $o_l$  is very popular), this taste (the favor for  $o_l$ ) is ordinary and it does not mean  $u_i$  and  $u_j$  are very similar. Therefore, its contribution to  $s_{ij}$  should be weakened. On the other hand, provided that a user  $u_i$  with small degree has collected an unpopular object  $o_l$  (the degree of  $o_l$  is very small), this taste should be very special, the contribution of the edge connecting  $u_i$  and  $o_l$  should be enlarged. It is not very meaningful if a user with large degree has selected a popular object, while if an unpopular object is selected by a small-degree user, this edge would contain rich information on personalized preference. Accordingly, the contribution of the edge connecting  $u_i$  and  $o_l$  should be negatively correlated with  $k_{u_i}k_{o_l}$ . We assume a certain amount of resource (e.g. recommendation power) is associated with each user, and the weight  $s_{ij}$  represents the proportion of the resource  $u_j$  would like to distribute to  $u_i$ .

Lind *et al.* presented a cycle measurement to investigate the clustering property in bipartite network.<sup>26,27</sup> The standard definition of clustering coefficient  $C_3$  is the fraction between the number of triangles observed in one network out from the total number of possible triangles which may appear. The clustering coefficient  $C_3(i)$  for node  $i$  is

$$C_3(i) = \frac{2E_i}{k_i(k_i - 1)}, \quad (4)$$

where  $E_i$  is the number of triangles observed, i.e. the number of connections among the  $k_i$  neighbors. Similarly to  $C_3(i)$ , a cluster coefficient  $C_4(i)$  with squares is the quotient between the number of squares and the total number of possible squares. For a given node  $i$ , the number of observed squares is given by the number of common neighbors among its neighbors, while the total number of possible squares is given by the sum over each pair of neighbors of the product between their degrees, after subtracting the common node  $i$  and an additional one if they are connected. Explicitly this clustering coefficient reads<sup>26</sup>

$$C_4 = \frac{\sum_{x=1}^{k_i} \sum_{y=x+1}^{k_i} q_i(x, y)}{\sum_{x=1}^{k_i} \sum_{y=x+1}^{k_i} [a_i(x, y) + q_i(x, y)]}, \quad (5)$$

where  $x$  and  $y$  label neighbors of node  $i$ ,  $q_i(x, y)$  are the number of common neighbors between  $x$  and  $y$  and  $a_i(x, y) = (k_x - \eta_i(x, y))(k_y - \eta_i(x, y))$  with  $\eta_i(x, y) = 1 - q_i(x, y) + \theta_{xy}$  and  $\theta_{xy} = 1$  if neighbors  $x$  and  $y$  are connected with each other and 0 otherwise. The presented definitions have brought us a new way to study the performance of the recommendation algorithms. Based on the correlations between the users and objects, we constructed an improved collaborative filtering algorithm. The algorithm could be implemented in the following way. Following a network-based resource-allocation process where each user distributes his/her initial resource to all the objects he/she has collected, and then each object sends back what it has received to all the users who collected it, considering the correlation between users and objects, the weight  $s_{ij}$  (the fraction of initial resource

Table 1. Algorithmic performance for *MovieLens* data. The precision, diversity and popularity are corresponding to  $L = 50$ .

Algorithms	$\langle r \rangle$	$S$	$\langle k \rangle$
GRM	0.1390	0.398	259
CF	0.1168	0.549	246
ICF	0.1156	0.630	229
ZhouM	0.0820	0.793	175
MCF	0.0998	0.692	218

$u_j$  eventually gives to  $u_i$ ) can be expressed as

$$s_{ij} = \frac{1}{k_{u_j}} \sum_{l=1}^m \frac{a_{li}(k_{u_j} k_{o_l})^\lambda \cdot a_{lj}(k_{u_i} k_{o_l})^\lambda}{k_{o_l}}, \quad (6)$$

where  $\lambda$  is a tunable parameter controlling the effect of degree correlation.

## 5. Numerical Results

A benchmark dataset, namely MovieLens, is used to test the above algorithm, which consists of 1682 movies (objects) and 943 users. The users vote movies by discrete ratings from one to five. We applied a coarse-graining method: A movie is set to be collected by a user only if the given rating is larger than 2. The original data contains  $10^5$  ratings, 85.52% of which are larger than 2, that is, the user-object (user-movie) bipartite network after the coarse graining contains 82 520 edges.

Figure 1 reports the algorithmic accuracy as a function of  $\lambda$ . The curve has a clear minimum around  $\lambda = -0.96$ , which strongly supports the above discussion

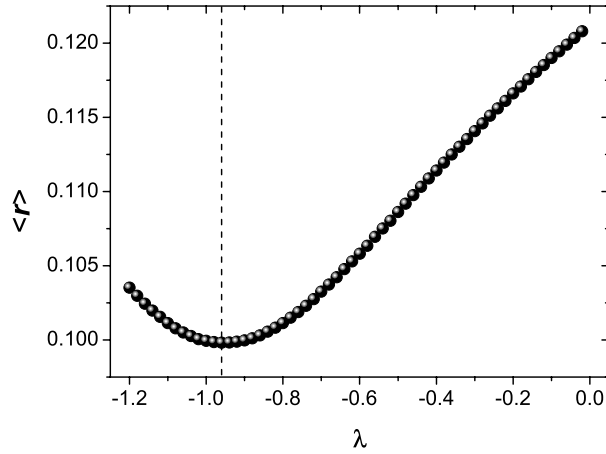


Fig. 1. The average ranking score  $\langle r \rangle$  vs  $\lambda$  for the algorithm. The optimal  $\lambda_{\text{opt}}$ , corresponding to the minimal  $\langle r \rangle = 0.0998$ , is  $\lambda_{\text{opt}} = -0.96$ . When  $\lambda = 0$ , the algorithm degenerates to the accuracy of the CF based on the diffusion process. All the data points are averaged over 10 independent runs with different data-set divisions.

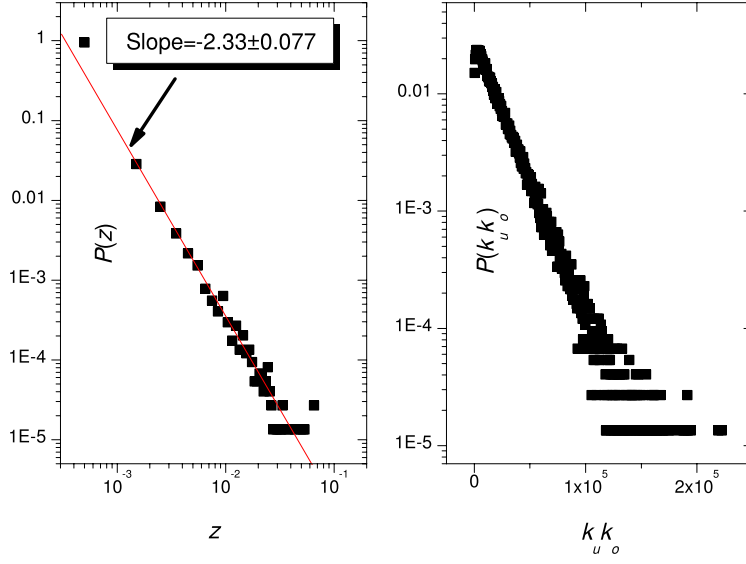


Fig. 2. Distributions of the user and object degrees, the left one demonstrates the distribution at the optimal parameter  $\lambda_{\text{opt}} = -0.96$ , where  $z = (k_u k_o)_{\text{opt}}^\lambda$  and the right one corresponds to the one when  $\lambda = 1.0$ .

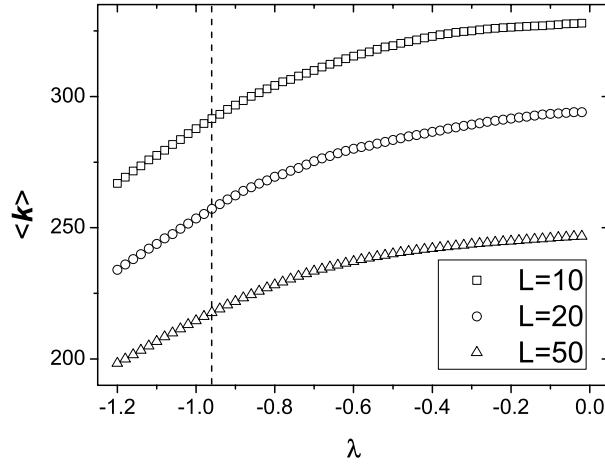


Fig. 3. Average degrees of recommended objects,  $\langle k \rangle$  vs  $\lambda$ . Squares, circles and triangles represent lengths  $L = 10, 20$ , and  $50$ , respectively. All the data points are averaged over 10 independent runs with different data-set divisions.

that to depress the influence of the users or objects with large degrees could enhance the accuracy. Compared with the routine case ( $\lambda = 0$ ), the average ranking score can be reduced by 18.19% at the optimal case, which indeed is a great improvement. Table 1 has demonstrated the accuracy obtained by several algorithms, from which one can find all three metrics, including the accuracy, average object degree of

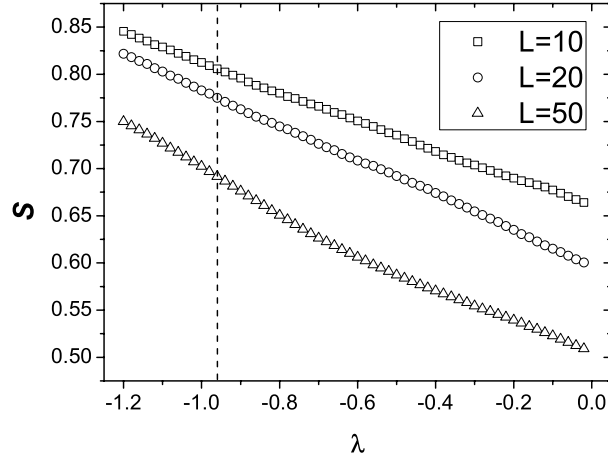


Fig. 4. The diversity  $S$  vs  $\lambda$ . Squares, circles and triangles represent the lengths  $L = 10, 20$ , and  $50$ , respectively. All the data points are averaged over 10 independent runs with different data-set divisions.

the recommended objects and diversity, of MCF is much better than GRM and the traditional CF algorithm, which is much larger than the ones obtained by ZhouM. In Table 1, ICF and ZhouM are abbreviations of the algorithms proposed in Refs. 7 and 22, MCF is an abbreviation of the algorithm presented in this paper. The parameters in ICF and ZhouM are set as the ones corresponding to the lowest ranking scores (for ICF,  $\lambda_{\text{opt}} = 1.9$ ; for ZhouM,  $\beta_{\text{opt}} = -0.80$ ). Each number presented in this table is obtained by averaging over 10 runs, each of which has an independently random division of training set and probe. It should be noted that ZhouM must use the second-order correlation information to eliminate the redundant information, which increases the computation complexity greatly and makes them hard for application. In addition, the product distribution of the user and object degrees is investigated in Fig. 2. One can see from the left that, different from the exponential distribution form of the product when  $\lambda = 1$ , the product distribution has a power-law form whose exponential is equal to  $-2.33 \pm 0.077$  at the optimal case. Figure 3 reports the average degrees of all recommended movies as a function of  $\lambda$ . When  $\lambda < 0$ ,  $\langle k \rangle$  is positively correlated with  $\lambda$ , thus to depress the influence of edges connecting active users and popular objects gives more opportunity to the unpopular objects, which is consistent with our expectation. Figure 4 exhibits a negative correlation between  $S$  and  $\lambda$ , indicating that to depress the influence of the edges connecting active users and popular objects makes the recommendations more personalized. When  $L = 10$ , the diversity  $S$  is increased from 0.661 (corresponding to the case when  $\lambda = 0$ ) to 0.806 (corresponding to the optimal case  $\lambda = -0.96$ ), improved by 21.90%.



## 6. Conclusions and Discussions

In this paper, a modified CF algorithm is presented by investigating the degree correlation effect of users and objects. The numerical results show that by depressing the influence of edges connecting active users and popular objects, the algorithmic accuracy, measured by the average ranking score, can be improved by 18.19%. At the optimal case,  $\lambda = -0.96$ , the influences of the edges, whose user and object degree product are small, are strengthened, more surprisingly, the product distribution has a power-law form. Although we cannot explain clearly what's the relationship between the power-law distribution and the optimal accuracy, we believe there exists a mathematical correlation between these two elements. Besides accuracy, two significant criteria of algorithmic performance, popularity and diversity, are also taken into account. A good recommendation algorithm should not only have higher accuracy, but also to help the users uncover the hidden information, corresponding to those objects with low degrees. Therefore, the average object degree of the recommended objects is a meaningful measurement for a recommendation algorithm. In addition, a personalized recommender system should provide each user personalized recommendations according to his/her interests and habits, therefore, the diversity of recommendations plays a crucial role to quantify the personalization. The numerical results show that the presented algorithm outperforms the standard CF in all three criteria, accuracy, popularity and diversity.

Although the accuracy is not as good as the ones obtained by the algorithms,<sup>8,22</sup> it does not use the second-order similarity, therefore, the algorithm complexity is very limited, which makes the presented algorithm more practical. Since the power computation takes much more time than multiplication, this algorithm would take a longer time to get the user similarities. Throughout the numerical simulation results, we could find that the optimal  $\lambda_{\text{opt}}$  is close to  $-1$ . When  $\lambda = -1$ , the corresponding  $\langle r \rangle = 0.0995$ , which is also improved by 18.07%, and the average object degree and diversity are getting even better, where the diversity  $S$  has been improved by 23.40%. Therefore, in real applications, the parameter could be set as  $-1$ , which ensures that the algorithmic complexity is the same as a parameter-free CF. The presented algorithm could improve the algorithmic accuracy by considering the effect of the user and object degree correlation, and the computation complexity is much smaller than the one of the traditional CF algorithm, however, to different data set, how to find the optimal parameter,  $\lambda_{\text{opt}}$ , has hindered its application. To an empirical dataset, there are at least two ways to find the optimal parameter. The first one is to construct a small sample dataset, and find the correlation between the parameter and the algorithmic performance, then find the optimal parameter. The other possible way is to analyze the degree product distribution of the user and object. In this paper, we find that, at the optimal case, the distribution has been changed from the exponential form to the power-law, although we cannot give a clear reason to explain it, we believe the algorithmic performance could be improved by analyzing the exponent of the product distribution.

The further work should focus on how to find the correlation between the structural characteristics, such as  $C_3$  and  $C_4$ , and the algorithmic performance. How to automatically find out relevant information for diverse users is a long-standing challenge in the modern information science, the presented algorithm could also be used to find the relevant reviewers for the scientific papers or funding applications,<sup>28,29</sup> and the link prediction in social and biological networks.<sup>30</sup> We believe the current work can enlighten readers in this promising direction.

## Acknowledgments

We acknowledge GroupLens Research Group for providing us the data. This work is partially supported by the National Basic Research Program of China (No. 2006CB705500), the National Natural Science Foundation of China (Nos. 10905052, 70901010, 10635040, 60973069, 90924011), the Swiss National Science Foundation (Project 205120-113842), Shanghai Leading Discipline Project (No. S30501) and Shanghai Development of Undergraduate Education Base III-Electronic Commerce.

## References

1. G.-Q. Zhang *et al.*, *New J. Phys.* **10**, 12307 (2008).
2. P. Resnick and H. R. Varian, *Commun. ACM* **40**, 56 (1997).
3. G. Adomavicius and A. Tuzhilin, *IEEE Trans. Knowl. Data Eng.* **17**, 734 (2005).
4. J. B. Schafer, J. A. Konstan and J. Riedl, *Data Min. Knowl. Disc.* **5**, 115 (2001).
5. J. L. Herlocker *et al.*, *ACM Trans. Inform. Syst.* **22**, 5 (2004).
6. J. A. Konstan *et al.*, *Commun. ACM* **40**, 77 (1997).
7. J.-G. Liu, B.-H. Wang and Q. Guo, *Int. J. Mod. Phys. C* **20**, 285 (2009).
8. J.-G. Liu *et al.*, *Physica A* **389**, 881 (2010).
9. J.-G. Liu *et al.*, *Int. J. Mod. Phys. C* **20**, 1925 (2009).
10. D. Sun *et al.*, *Phys. Rev. E* **80**, 017101 (2009).
11. M. Balabanović and Y. Shoham, *Commun. ACM* **40**, 66 (1997).
12. M. J. Pazzani, *Artif. Intell. Rev.* **13**, 393 (1999).
13. Y. Gao, H. Luo and J. Fan, *Lect. Notes in Comp. Sci.* **5371**, 217 (2009).
14. H. Luo *et al.*, *Lect. Notes in Comp. Sci.* **5371**, 459 (2009).
15. M. Pazzani and D. Billsus, *Machine Learning* **27**, 313 (1997).
16. C. Basu, H. Hirsh and W. Cohen, *Technical Report WS-98-08* (AAAI Press, 1998), p. 714.
17. N. Good *et al.*, *Proc. Conf. Am. Assoc. Artif. Intell.* 439 (1999).
18. J. B. Schafer *et al.*, *Lect. Notes Comput. Sci.* **4321**, 291 (2007).
19. Y.-C. Zhang *et al.*, *Europhys. Lett.* **80**, 68003 (2008).
20. T. Zhou *et al.*, *Phys. Rev. E* **76**, 046115 (2007).
21. Y.-C. Zhang, M. Blattner and Y.-K. Yu, *Phys. Rev. Lett.* **99**, 154301 (2007).
22. T. Zhou *et al.*, *New J. Phys.* **11**, 123008 (2009).
23. T. Sorenson, *Biol. Skr.* **5**, 1 (1948).
24. G. Salton and M. J. McGill, *Introduction to Modern Information Retrieval* (McGraw-Hill, New York, 1983).
25. T. Zhou *et al.*, *Europhys. Lett.* **81**, 58004 (2008).
26. P. G. Lind, M. C. González and H. J. Herrmann, *Phys. Rev. E* **72**, 056127 (2005).

- 27. P. G. Lind and H. J. Herrmann, *New J. Phys.* **9**, 228 (2007).
- 28. J.-G. Liu, Y.-Z. Dang and Z.-T. Wang, *Physica A* **366**, 578 (2006).
- 29. J.-G. Liu *et al.*, *Physica A* **377**, 302 (2007).
- 30. T. Zhou, L. Lü and Y.-C. Zhang, *Eur. Phys. J. B* **71**, 623 (2009).